

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the above-identified application.

Listing of Claims:

1 -63. (Canceled)

64. (New) A method for generating a translation of subject code into target code, comprising:

identifying a block of self-modifying subject code within a block of subject code;

identifying one or more blocks of non-self-modifying code within the block of subject code;

partitioning the block of subject code, comprising:

generating a first partition in the subject code such that the first partition

includes the identified block of self-modifying subject code;

generating one or more additional partitions in the subject code such that

each additional partition includes one of the identified blocks of

non-self-modifying subject code, respectively, wherein subject

code in each of the first partition and the one or more partitions do

not overlap;

translating the first partition into a first block of target code;

translating the one or more additional partitions into a corresponding number of

additional blocks of target code; and

combining the first block of target code and the additional blocks of target code
to produce a translated target code.

65. (New) The method of claim 64, further comprising:
detecting a modification in the block of self-modifying subject code;
translating the first partition, including the modification, to produce a modified
block of target code; and
combining the modified block of target code and the additional blocks of target
code to produce a modified translated target code.

66. (New) The method of claim 64, further comprising:
detecting a modification in the block of self-modifying subject code;
detecting that the modification has modified a block of code corresponding to a
second partition of the one or more additional partitions;
modifying the first partition by adding the block of code corresponding to the
second partition to produce a modified first partition;
modifying the second partition by removing the block of code corresponding to
the second partition to produce a modified second partition;
translating the first and second modified partitions to produce a first and second
modified blocks of target code, respectively; and
combining the first and second modified blocks of target code and the additional
blocks of target code, but not a block of target code corresponding to the
second partition, to produce a modified translated target code.

67. (New) The method of claim 64, further comprising:

identifying two partitions of the first and additional partitions that are stored adjacently in a memory and that have characteristics that allow them to be combined;

aggregating the two partitions into a single combined partition;

translating the combined partition to produce a modified combined block of target code; and

combining the first block of target code, the additional blocks of target code and the modified combined block of target code, without blocks of target code corresponding to the two partitions, to produce a modified translated target code

68. (New) The method of claim 64, further comprising:

identifying a control flow instruction in the block of self-modifying subject code or in the one or more blocks of non-self-modifying subject code; and

inserting an exit translation structure, representative of the control flow instruction, into a block of code in which the control flow instruction was identified.

69. (New) The method of claim 68, further comprising:

identifying a block of code as a target of the control flow instruction; and

inserting an entry translation structure into the block of code identified as the target of the control flow instruction.

70. (New) The method of claim 69, further comprising:

inserting, into the block of code in which the control flow instruction was identified, a first border guard corresponding to the exit translation structure; and

inserting, into the block of code identified as the target of the control flow instruction, a second border guard corresponding to the entry translation structure.

71. (New) A computer programming product, comprising:

a computer-readable memory, executable on a processor; and

logic, stored on the computer-readable memory and executed on the processor, for:

identifying a block of self-modifying subject code within a block of subject code;

identifying one or more blocks of non-self-modifying code within the block of subject code;

partitioning the block of subject code, comprising:

generating a first partition in the subject code such that the first partition includes the identified block of self-modifying subject code;

generating one or more additional partitions in the subject code such that each additional partition includes one of the identified blocks of non-self-modifying subject code,

respectively, wherein subject code in each of the first partition and the one or more partitions do not overlap;
translating the first partition into a first block of target code;
translating the one or more additional partitions into a corresponding number of additional blocks of target code; and
combining the first block of target code and the additional blocks of target code to produce a translated target code.

72. (New) The computer programming product of claim 71, the logic further comprising logic for:

detecting a modification in the block of self-modifying subject code;
translating the first partition, including the modification, to produce a modified block of target code; and
combining the modified block of target code and the additional blocks of target code to produce a modified translated target code.

73. (New) The computer programming product of claim 71, the logic further comprising logic for:

detecting a modification in the block of self-modifying subject code;
detecting that the modification has modified a block of code corresponding to a second partition of the one or more additional partitions;
modifying the first partition by adding the block of code corresponding to the second partition to produce a modified first partition;

modifying the second partition by removing the block of code corresponding to the second partition to produce a modified second partition;
translating the first and second modified partitions to produce a first and second modified blocks of target code, respectively; and
combining the first and second modified blocks of target code and the additional blocks of target code, but not a block of target code corresponding to the second partition, to produce a modified translated target code.

74. (New) The computer programming product of claim 71, the logic further comprising logic for:

identifying two partitions of the first and additional partitions that are stored adjacently in a memory and that have characteristics that allow them to be combined;
aggregating the two partitions into a single combined partition;
translating the combined partition to produce a modified combined block of target code; and
combining the first block of target code, the additional blocks of target code and the modified combined block of target code, without blocks of target code corresponding to the two partitions, to produce a modified translated target code

75. (New) The computer programming product of claim 71, the logic further comprising logic for:

identifying a control flow instruction in the block of self-modifying subject code
or in the one or more blocks of non-self-modifying subject code; and
inserting an exit translation structure, representative of the control flow
instruction, into a block of code in which the control flow instruction was
identified.

76. (New) The computer programming product of claim 75, the logic further comprising logic for:

identifying a block of code as a target of the control flow instruction; and
inserting an entry translation structure into the block of code identified as the target of the
control flow instruction.

77. (New) An apparatus, comprising:

a processor;
a computer-readable memory coupled to the processor; and
logic, stored on the computer-readable memory and executed on the processor,
for:
identifying a block of self-modifying subject code within a block of
subject code;
identifying one or more blocks of non-self-modifying code within the
block of subject code;

partitioning the block of subject code, comprising:

generating a first partition in the subject code such that the first partition includes the identified block of self-modifying subject code;

generating one or more additional partitions in the subject code such that each additional partition includes one of the identified blocks of non-self-modifying subject code, respectively, wherein subject code in each of the first partition and the one or more partitions do not overlap;

translating the first partition into a first block of target code;

translating the one or more additional partitions into a corresponding number of additional blocks of target code; and

combining the first block of target code and the additional blocks of target code to produce a translated target code.

78. (New) The apparatus of claim 77, the logic further comprising logic for:

detecting a modification in the block of self-modifying subject code;

translating the first partition, including the modification, to produce a modified block of target code; and

combining the modified block of target code and the additional blocks of target code to produce a modified translated target code.

79. (New) The apparatus of claim 77, the logic further comprising logic for:

- detecting a modification in the block of self-modifying subject code;
- detecting that the modification has modified a block of code corresponding to a second partition of the one or more additional partitions;
- modifying the first partition by adding the block of code corresponding to the second partition to produce a modified first partition;
- modifying the second partition by removing the block of code corresponding to the second partition to produce a modified second partition;
- translating the first and second modified partitions to produce a first and second modified blocks of target code, respectively; and
- combining the first and second modified blocks of target code and the additional blocks of target code, but not a block of target code corresponding to the second partition, to produce a modified translated target code.

80. (New) The apparatus of claim 77, the logic further comprising logic for:

- identifying two partitions of the first and additional partitions that are stored adjacently in a memory and that have characteristics that allow them to be combined;
- aggregating the two partitions into a single combined partition;
- translating the combined partition to produce a modified combined block of target code; and
- combining the first block of target code, the additional blocks of target code and the modified combined block of target code, without blocks of target code

corresponding to the two partitions, to produce a modified translated target code

81. (New) The apparatus of claim 77, the logic further comprising logic for:
identifying a control flow instruction in the block of self-modifying subject code
or in the one or more blocks of non-self-modifying subject code; and
inserting an exit translation structure, representative of the control flow instruction, into a block of code in which the control flow instruction was identified.

82. (New) The apparatus of claim 81, the logic further comprising logic for:
identifying a block of code as a target of the control flow instruction; and
inserting an entry translation structure into the block of code identified as the target of the control flow instruction.